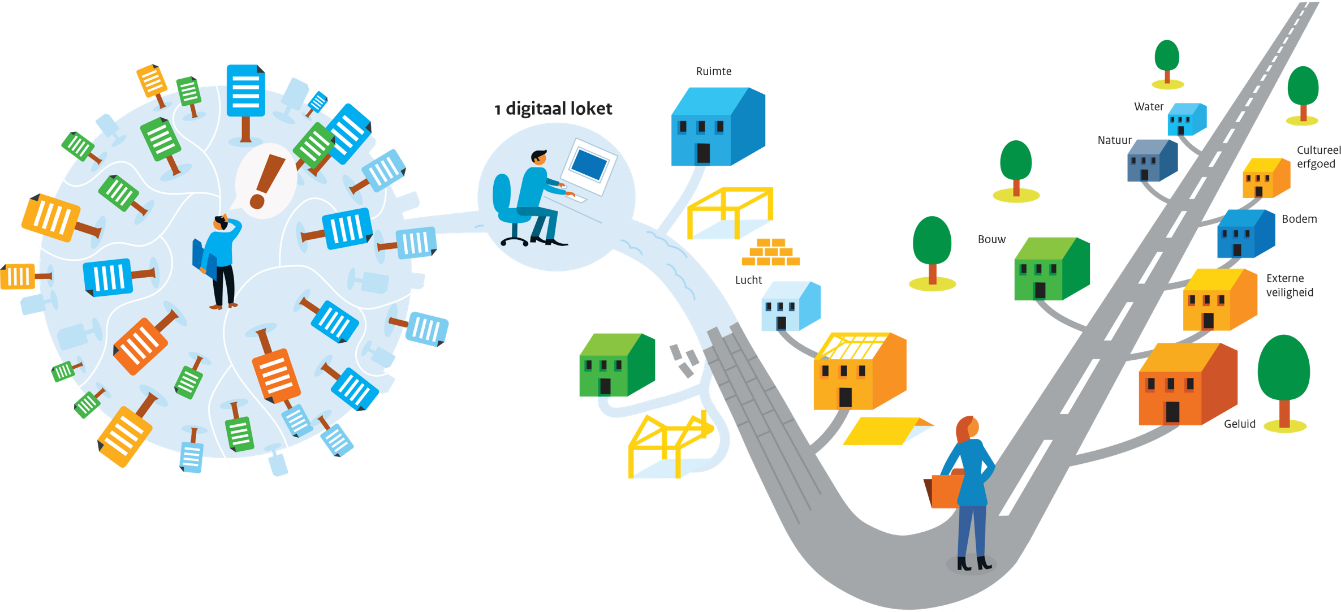
Deelprogramma  
Digitaal Stelsel Omgevingswet

Kaderstellende notities

URI-strategie

Versie 1.57 Concept 15-05-2017



***Tim Berners-Lee in 1999 over een Uniform Resource Identifier (URI):***

*‟Een soort ‘adres’ dat uniek is en gebruikt wordt om iedere resource op het web te identificeren. Het wordt gewoonlijk een URL genoemd.„*

Colofon

Titel : URI-strategie

Versie : 1.57 Concept

Datum : 15-05-2017

Opdrachtgever : Programma Implementatie Omgevingswet

Opdrachtnemer : Deelprogramma DSO

Auteurs : Stephen Oostenbrink

Marco Brattinga

Tony Sloos

Contactpersonen : Tony Sloos

*Domeinarchitect Informatie en Kernfuncties*

+31 6 1125 2597

[tony.sloos@rws.nl](mailto:tony.sloos@rws.nl)

Versiehistorie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versie | Status | Datum | Auteur(s) | Toelichting |
| 1.35 | Concept | 29-11-2016 | M. Brattinga | Initiële versie. |
| 1.40 | Concept | 30-11-2016 | M. Brattinga | Review verwerkt van SO. |
| 1.50 | Concept | 06-12-2016 | M. Brattinga | Review verwerkt van SO. |
| 1.51 | Concept | 13-12-2016 | M. Brattinga | Review verwerkt van SO. |
| 1.52 | Concept | 20-12-2016 | M. Brattinga | Review verwerkt van BC, EM, FT, LvB, NP, SM, TS. |
| 1.53 | Concept | 23-01-2017 | M. Brattinga | Review verwerkt van SO. |
| 1.54 | Concept | 24-01-2017 | S. Oostenbrink | Omgezet naar standaard template. |
| 1.55 | Concept | 20-02-2017 | S. Oostenbrink | Plek versienummer in URI hersteld. Was per abuis aangepast. |
| 1.56 | Concept | 12-04-2017 | A.J. Sloos | Review Peter Visser verwerkt. |
| 1.57 | Concept | 15-05-2017 | A.J. Sloos | Kaders en lijst met eisen toegevoegd. Input externe consultatie verwerkt. |

Goedkeuring

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functie | Naam | Versie | Datum | Handtekening |
| Programma Directeur Implementatie Omgevingswet  (namens de Programma Raad) | Ineke van der Hee |  |  |  |
| Programma Manager PDSO | Bert Uffen |  |  |  |
| Lead architecten programma | Victorine Binkhorst |  |  |  |

Distributie

|  |  |  |
| --- | --- | --- |
| Functie/Orgaan | Versie | Opmerkingen |
| Opdrachtgevend Beraad Implementatie Omgevingswet |  |  |
| Programma Raad Implementatie Omgevingswet |  |  |
| Programma Team |  |  |
| Programma Architectuur Team | 1.55, 1.56 |  |
| Architectuur Team Overleg | 1.55, 1.56 |  |
| Projectmanager |  |  |
| Strategische Ontwikkelpartners | 1.55, 1.56 |  |

Review

| Naam | Versies |
| --- | --- |
| Andre Batenburg (AB), BLA Provinces | 1.55, 1.56 |
| Jan Langeveld (JL), BLA Gemeenten | 1.55, 1.56 |
| Marco Brattinga (MB), Projectarchitect PR06 | 1.55, 1.56 |
| Nico Plat (NP), Projectarchitect PR12 | 1.51, 1.56 |
| Pascal Thuis (PT), Projectarchitect PR04 | 1.51, 1.56 |
| Paul de Frankrijker (PF), BLA Waterschappen | 1.55, 1.56 |
| Peter Visser (PV), BLA Rijk | 1.55, 1.56 |
| Rene Kint (RK), Stelselarchitect | 1.55, 1.56 |
| Silvion Moesan (SM), Projectarchitect PR02 | 1.51, 1.56 |
| Frank Terpstra (FT), Projectarchitect PR05 | 1.51, 1.56 |
| Erno Marks (EM), Ontwikkelaar PR05 | 1.51, 1.56 |
| Tony Sloos (TS), Domeinarchitect | 1.51, 1.56 |
| Bas Crompvoets (BC), Domeinarchitect | 1.51, 1.56 |
| Lennart van Bergen (LvB), Projectarchitect PR06 en PR30 | 1.51, 1.56 |

Inhoudsopgave

[1. Inleiding 6](#_Toc482685751)

[1.1. Doelgroep 6](#_Toc482685752)

[1.2. Doel 6](#_Toc482685753)

[1.3. Resultaat 6](#_Toc482685754)

[1.4. Afkortingen en begrippen 6](#_Toc482685755)

[1.5. Leeswijzer 6](#_Toc482685756)

[2. URI-strategie 7](#_Toc482685757)

[2.1. Pas toe of leg uit 7](#_Toc482685759)

[2.2. Resources, URI’s, URN’s en URI’s 7](#_Toc482685760)

[2.3. Basis opbouw URI’s 8](#_Toc482685761)

[2.4. Webpagina’s van gebruikerstoepassingen 9](#_Toc482685762)

[2.5. SOAP web services 10](#_Toc482685763)

[2.6. REST API’s 10](#_Toc482685764)

[2.7. Linked Data 11](#_Toc482685765)

[2.8. Nadere toelichting van de afzonderlijke componenten 12](#_Toc482685766)

[2.8.1. Autoriteit 12](#_Toc482685777)

[2.8.2. Onderdeel 13](#_Toc482685778)

[2.8.3. Beveiligingscontext 14](#_Toc482685779)

[2.8.4. Domein 14](#_Toc482685780)

[2.8.5. Type (Linked Data) 16](#_Toc482685781)

[2.8.6. Collectie en referentie 16](#_Toc482685782)

[2.9. Tijdreizen 17](#_Toc482685783)

[2.10. Versionering van SOAP web services en REST API’s 19](#_Toc482685784)

[2.11. Gewenst formaat 19](#_Toc482685785)

[2.12. Afspraken URI’s 20](#_Toc482685786)

[Bijlage A: Afkortingen 21](#_Toc482685787)

[Bijlage B: Begrippen 22](#_Toc482685788)

[Bijlage C: Eisen 23](#_Toc482685789)

[Bijlage D: Open Einden 26](#_Toc482685790)

# Inleiding

Dit document is kaderstellend voor de uniforme opbouw van URI’s. De uniforme opbouw van URI’s is een belangrijk instrument voor de uitwisseling van (meta)data met en binnen het DSO. URI’s zorgen voor de identificatie van resources die op basis van de Linked Data standaard worden ontsloten voor gebruik in gebruikerstoepassingen, API’s en services.

Aangezien de URI-strategie voor het DSO als geheel van belang is, is de URI-strategie een bijlage bij de Overall GAS (OGAS).

## Doelgroep

Dit document richt zich op programma en projecten en andere geïnteresseerden.

## Doel

Het doel van dit document is de kaders meegeven voor de uniforme opbouw van URI’s van het stelsel. Afwijken van deze kaders kan alleen in overleg met akkoord van de Stelsel Architectuur Board (SAB) van het DSO.

## Resultaat

Eenduidig beschrijving van de kaders waaraan de URI’s van het stelsel en de LVBB gehouden zijn.

## Afkortingen en begrippen

In de laatste versie van het document “DSO - Architectuur - Afkortingen Begrippen” staan de definities van begrippen en afkortingen die in dit kaderstellend document worden gehanteerd.

## Leeswijzer

Dit document is zelfstandig te lezen. Er zijn twee documenten met nadere uitwer­kingen van onderdelen, namelijk “DSO - Architectuur - API-strategie” en “DSO - Architectuur - Tijdreizen”.

# URI-strategie

Met de URI-strategie voor het DSO wordt alle informatie van het stelsel op een uniforme en samen­hangende manier vindbaar en toegankelijk. De URI-strategie schept duidelijkheid hoe URI’s opgebouwd moeten worden. URI’s bieden een mechanisme om naar resources te verwijzen ongeacht waar deze zich bevinden. URI’s zijn voor mensen leesbaar omdat dit de interpretatie en het leggen van relaties vereenvoudigd.

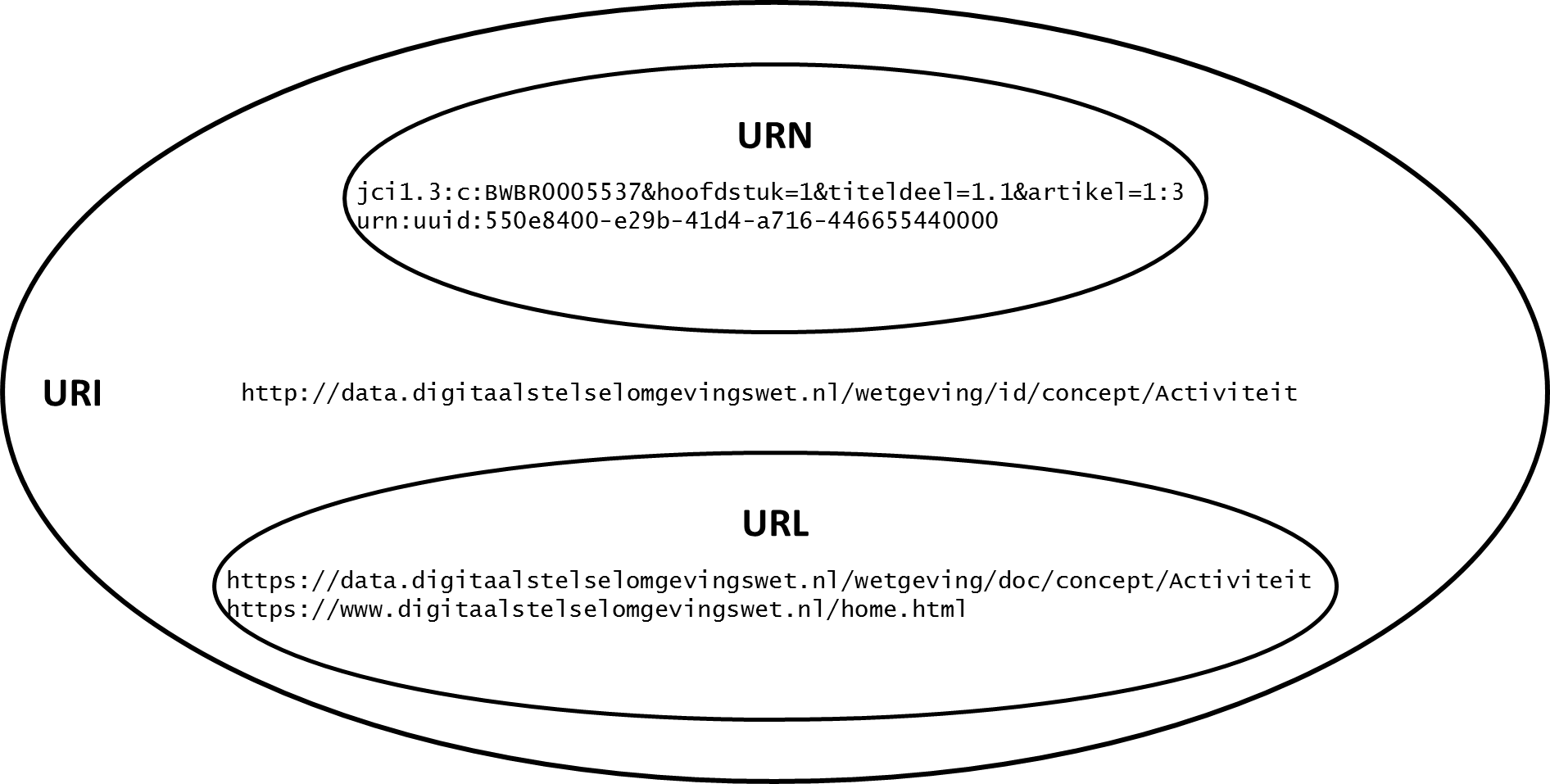


## Pas toe of leg uit

De URI-strategie in dit document betreft een “pas-toe-of-leg-uit”. Dit betekent dat stelselcomponenten bij publiceren van URI's geacht worden zich te houden aan de URI-strategie in dit document. Uiteraard is hierop ook het past-toe-of-leg-uit principe van toepassing. Een voorbeeld van zo’n leg-uit is een SOAP service in een COTS pakket dat niet aangepast kan worden.

## Resources, URI’s, URN’s en URI’s

Om deze notitie te begrijpen is het belangrijk om de begrippen URI, URN, URL en resource toe te lichten.



Een “resource” kan elk mogelijk ding zijn, zoals fysieke objecten (monument, straat, geografische locatie), abstracties (begrip, gegevenselement) alsook informatie-elementen (webpagina). URN’s, URI’s en URL’s gebruiken de term “resource”.

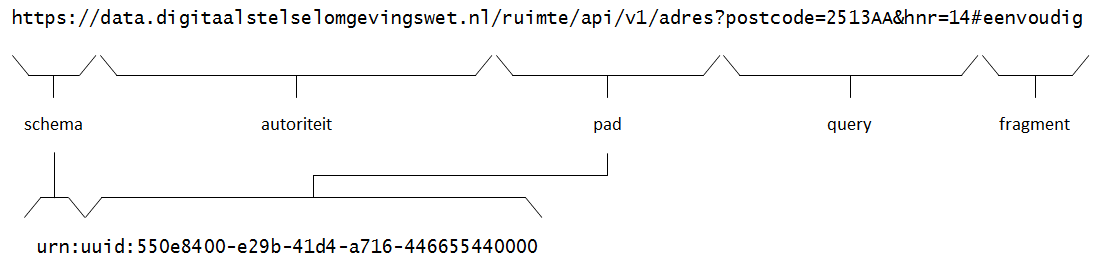
Een **Uniform Resource Identifier** (URI) beschrijft eenduidig de identificatie van een “resource”. Via de URI kan informatie over een dergelijke resources beschikbaar gesteld worden.

Een **Uniform Resource Name** (URN) beschrijft eenduidig de naam van een “resource”. Een URN is een URI, maar geen URL. Een URN is zonder inbedding in een URI of URL niet vindbaar op het internet.

Een **Uniform Resource Locator** (URL) beschrijft eenduidig de locatie van een “resource” en deze locatie is wel vindbaar op het internet. Een URL wordt gebruikt om anderen toegang te geven tot een bron. Binnen het DSO worden URL’s gebruikt voor zowel het Omgevingsloket en daarbinnen de gebruikerstoepassingen alsook voor API’s en services die via het Stelselknooppunt DSO (aka Knooppunt) worden aangeboden.

## Basis opbouw URI’s

Er wordt zoveel mogelijk gekozen voor uniformiteit bij de opbouw van URI’s om de herkenbaar­heid te verhogen en de leercurve te beperken. De basis opbouw van URI’s voor elk van deze categorieën volgt internet­standaard RFC3986:



|  |  |
| --- | --- |
| ☝ | 1. De basisopbouw van URI’s volgt internetstandaard RFC3986 |
| De basis opbouw van URI’s voor elk van deze categorieën volgt internetstandaard RFC3986.  De URI’s van het DSO kennen daarmee de volgende generieke opbouw:  <schema>“://”<autoriteit>“/”<pad>[“?”<query>]“/”[“#”<fragment>] |

Onderstaande tabel licht de verschillende componenten van de generieke URI opbouw kort toe. In de volgende paragrafen volgt een uitgebreide toelichting per categorie en onderdelen.

|  |  |
| --- | --- |
| Component | Toelichting |
| schema | Geeft het protocol aan waarmee de URI opgevraagd kan worden, bijvoorbeeld HTTP, HTTPS of FTP. |
| autoriteit | Wordt gebruikt om te bepalen welk stelselonderdeel de routering afhandelt:   1. Het Omgevingsloket routeert www.digitaalstelselomgevingswet.nl en digitaalstelselomgevingswet.nl; 2. Het Stelselknooppunt DSO routeert data.digitaalstelselomgevingswet.nl voor API’s, web services en linked data. |
| pad | Het unieke pad dat een resource identificeert of een locatie van een resource aangeeft. Hierbij zijn de onderdelen van het pad gescheiden door een slash (“/”). De slash wordt gebruikt om hiërarchie aan te geven. |
| query | Het query deel bevat niet-hiërarchische data die samen met het pad de resource identificeert. Het gaat hier vaak om parameters die als filter werken. |
| fragment | Een fragment identificatie is een identificatie van een resource die ondergeschikt is aan een andere primaire resource. |

Er worden vier verschillende resourcecategorieën onderscheiden die elk via URI’s zijn te benaderen en naast algemene afspraken eigen specifieke afspraken kennen.

De volgende vier resourcecategorieën worden onderscheiden:

1. Webpagina’s van gebruikerstoepassingen;
2. SOAP web services[[1]](#footnote-1), [[2]](#footnote-2);
3. REST API’s[[3]](#footnote-3);
4. Linked Data[[4]](#footnote-4): data die geïdentificeerd en opgevraagd wordt d.m.v. URI’s.

|  |  |
| --- | --- |
| ☝ | 1. In de basisopbouw van URI zijn de onderkende resourcecategorieën duidelijk herkenbaar |
| De volgende vier verschillende categorieën worden minimaal onderscheiden:   1. Webpagina’s van gebruikerstoepassingen; 2. SOAP web services; 3. REST API’s; 4. Linked Data. |

## Webpagina’s van gebruikerstoepassingen

In het geval van een webpagina is het deel vanaf <pad> verder opgedeeld in de volgende componenten:

<onderdeel>[“/”<beveiliging>]”/”<pagina>[“?”<query>][“#”<fragment>]

|  |  |
| --- | --- |
| ☝ | 1. De URI van een webpagina identificeert het verantwoordelijke stelselonderdeel |
| Het URI-component <onderdeel> identificeert het stelselonderdeel dat verantwoordelijk is voor de betreffende webpagina. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een webpagina specificeert optioneel een expliciete beveiligingscontext |
| Toepassingen kunnen functionaliteiten bieden die een aparte beveiligingscontext vereisen. Deze beveiligingscontext dient er bijvoorbeeld voor om onderscheid te maken tussen gebruikersfunctionaliteit en beheerfunctionaliteit. Als een afnemer toegang heeft tot het ene en niet het andere dan dienen deze gescheiden te zijn. Hetzelfde principe als een UI als je geen toegang hebt tot functionaliteit dan zie je die functionaliteit ook niet. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een webpagina identificeert de pagina binnen het stelselonderdeel |
| Het URI-component <pagina> identificeert de webpagina binnen het stelselonderdeel. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een webpagina specificeert optioneel een reeks query-parameters |
| Het URI-component <query> specificeert een optionele reeks query-parameters. Het gebruik zal afhangen van de specifieke gebruikerstoepassing. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een webpagina specificeert optioneel een fragment binnen de pagina |
| Het URI-component <fragment> specificeert een optionele verwijzing naar een sectie binnen de webpagina. De browser zal automatisch scrollen naar het specifieke fragment binnen de webpagina. |

## SOAP web services

In het geval van een SOAP web-service is het deel vanaf <pad> verder opgedeeld in de volgende componenten:

<onderdeel>[“/”<beveiliging>]”/service/”<versie>“/”<service>

|  |  |
| --- | --- |
| ☝ | 1. De URI van een SOAP web-service identificeert het verantwoordelijke stelselonderdeel |
| Het URI-component <onderdeel> identificeert het stelselonderdeel dat verantwoordelijk is voor de betreffende SOAP web service. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een SOAP web-service specificeert optioneel een expliciete beveiligingscontext |
| SOAP web services kunnen functionaliteiten bieden die een aparte beveiligingscontext vereisen. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een SOAP web-service specificeert het major versienummer van de service |
| Het URI-component <versie> specificeert het versienummer van de service. Opbouw is “v” gevolgd door een nummer. Dit is de major versienummer van de service. Het versienummer begint bij 1 en wordt met 1 opgehoogd voor elke major release waar het koppelvlak wijzigt (niet backward compatible). Minor versienummers staan in het bericht zelf, maar niet in de namespace omdat dit backward compatible is. De namespace bevat alleen de major versienummer. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een SOAP web-service identificeert de service binnen het stelselonderdeel |
| Het URI-component <service> identificeert de service binnen het stelselonderdeel. Aangezien in SOAP parameters van services onderdeel zijn van het bericht, zal in het geval van SOAP geen sprake zijn van een <query> of <fragment> deel. |

## REST API’s

In het geval van een REST API[[5]](#footnote-5) is het deel vanaf <pad> verder opgedeeld in de volgende componenten:

<onderdeel>[“/”<beveiliging>]”/api/<versie>/”<collectie>[“/”<referentie>][“?”<query>]

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API identificeert het verantwoordelijke stelselonderdeel |
| Het URI-component <onderdeel> identificeert het stelselonderdeel dat verantwoordelijk is voor de betreffende SOAP web service. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API specificeert optioneel een expliciete beveiligingscontext |
| REST API’s kunnen functionaliteiten bieden die een aparte beveiligingscontext vereisen. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API specificeert het major versienummer van de API |
| Het URI-component <versie> specificeert het versienummer van de API. Opbouw is “v” gevolgd door een nummer. Dit is de major versienummer van de API. Het versienummer begint bij 1 en wordt met 1 opgehoogd voor elke major release waar het koppelvlak wijzigt (niet backward compatible). Minor versienummers staan in het bericht zelf. Het toevoegen van een eindpunt of een niet verplichte attribuut aan de payload zijn voorbeelden van wijzigingen die backward compatible zijn. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API identificeert de collectie binnen het stelselonderdeel |
| Het URI-component <collectie> identificeert de collectie waaruit gegevens worden teruggegeven. Dit kan een bestaande dataset zijn (bijvoorbeeld “/gebouwen”), maar ook een collectie die dynamisch wordt opgebouwd (bijvoorbeeld “/adressen”). In dit laatste geval zou je ook kunnen spreken over een “service”, vanuit een interactie-gedreven API. Collectie is meervoud indien meer dan één resultaat kan worden teruggegeven en enkelvoud als het resultaat altijd precies één resource betreft. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API specificeert optioneel een verwijzing |
| Het URI-component <referentie> specificeert optioneel een verwijzing naar de identificatie van een resource binnen de collectie. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van een REST API specificeert optioneel een reeks query-parameters |
| Het URI-component <query> specificeert optioneel aanvullende parameters die als filter werken. Dit is vooral handig als geen <referentie> is opgegeven, bijvoorbeeld: /adressen?postcode= 2513AA&hnr=14. Zie voor een volledige specificatie de DSO API-strategie. |

## Linked Data

Eén van de Linked Data principes is dat elke resource een eigen URI krijgt. Een dergelijke URI is vergelijkbaar met de primaire sleutel in een Relationele Database. Anders dan bij een relationele database is deze identificatie uniek op een wereldwijde schaal. Bij relationele databases is een identificatie vaak slechts uniek binnen de tabel waar deze wordt gebruikt. Het vormen van een dergelijke wereldwijd unieke identificatie wordt het “munten van een URI” genoemd.

In het geval van een Linked Data URI is het deel vanaf <pad> verder opgedeeld in de volgende componenten:

<domein>[“/”<lokaal>]“/”<type>“/”<collectie>“/”<referentie>  
<domein>[“/”<lokaal>]”/def/”<vocabulaire>“#”<fragment>

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data identificeert het DSO-domein van een resource |
| Het URI-component <domein> identificeert het DSO-domein waartoe een resource behoort. Een DSO-domein is een functionele identificatie en niet de identificatie van een (technische) stelselcomponent. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data identificeert optioneel een pad om lokale informatie te onderscheiden |
| Het URI-component <lokaal> identificeert een lokaal pad om onderscheid te maken naar lokale informatie, bijvoorbeeld een begrip van de gemeente Amersfoort[[6]](#footnote-6). |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data specificeert het type URI |
| Het URI-component <type> specificeert het type Linked Data URI. Mogelijke waarden zijn “id” voor identificatie van een resource, “doc” voor documentatie over een resource en “resource” voor de resource zelf. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data specificeert een vocabulaire |
| Het URI-component <vocabulaire> specificeert in het geval Linked Data URI-type “def” een vocabulaire. Direct na de type aanduiding volgt de term uit de vocabulaire (een klasse of eigenschap) |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data identificeert de collectie binnen de resource |
| Het URI-component <collectie> identificeert de collectie waartoe de resource behoort. Collectie dient hier vooral in relatie te worden gezien met referentie. De “referentie” is normaal gesproken de unieke identificatie van het object, maar vaak is zo’n unieke identificatie niet globaal uniek (denk bv. aan een BSN: dit 9-cijferige nummer kan ook iets anders zijn dan een referentie naar een persoon. Collectie is in dit geval dan dus “bsn” of zelfs “persoon” c.q. “burger”. Aangezien het resultaat van deze URI altijd precies één resource is, wordt collectie in enkelvoud geschreven. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data specificeert optioneel een verwijzing |
| Het URI-component <referentie> specificeert optioneel een verwijzing naar de identificatie van een resource binnen de collectie. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data specificeert optioneel een reeks query-parameters |
| Het URI-component <query> specificeert optioneel aanvullende parameters. Query wordt gebruikt bij het type “resource” om de identificatie van de resource op te geven: /resource?subject=<identificatie>. |

|  |  |
| --- | --- |
| ☝ | 1. De URI van Linked Data specificeert per vocabulaire tevens een klassenaam of eigenschapsnaam |
| Het URI-component <fragment> specificeert een klassenaam (indien het fragment met een hoofdletter begint) of de eigenschapsnaam (indien het fragment met een kleine letter begint). |

## Nadere toelichting van de afzonderlijke componenten



### Autoriteit

De opbouw van autoriteit is als volgt:

([www]|data).[<omgeving>.]“digitaalstelselomgevingswet.nl”

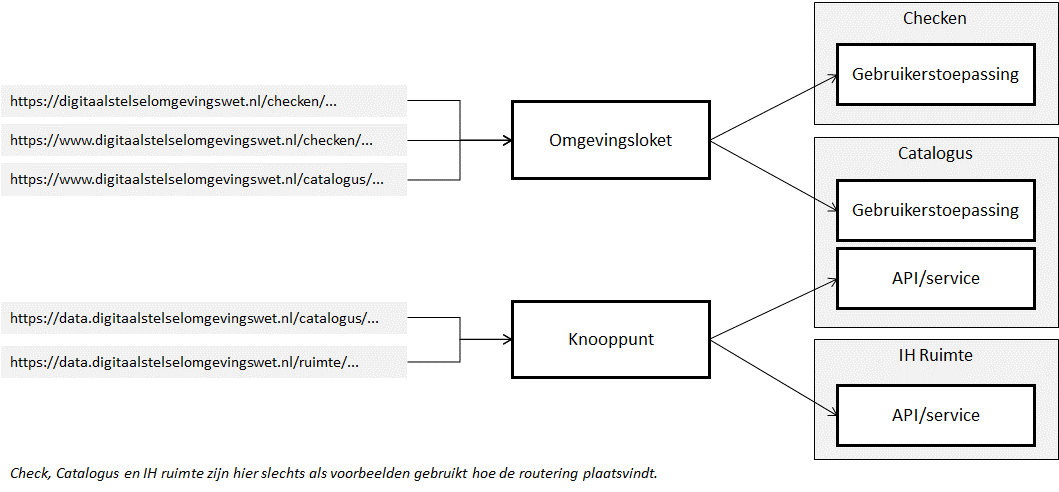
|  |  |
| --- | --- |
| ☝ | 1. Omgevingsloket URI’s definiëren de autoriteit binnen het sub-domein www |
| Voor het web portaal geldt als autoriteit www.<omgeving>.digitaalstelselomgevingswet.nl of <omgeving>.digitaalstelselomgevingswet.nl. |

|  |  |
| --- | --- |
| ☝ | 1. Stelselknooppunt URI’s definiëren de autoriteit binnen het sub-domein data |
| Voor het Stelselknooppunt DSO (services, API’s en data) geldt als autoriteit data.<omgeving>.digitaalstelselomgevingswet.nl. |

|  |  |
| --- | --- |
| ☝ | 1. URI’s definiëren optioneel de staging-omgeving binnen de autoriteit |
| De component <omgeving> verwijst naar de OTAP-omgeving, bijvoorbeeld www.test.digitaalstelselomgevingswet.nl. Omgeving kan één van de volgende waarden zijn:   * dev (ontwikkeling) * tst (test) * int (integratie) * acc (acceptatie) * pre (pre-productie)   Voor productie is deze component leeg, bijvoorbeeld www.digitaalstelselomgevingswet.nl. |

### Onderdeel

Er is altijd één stelselcomponent verantwoordelijk voor het aanbieden van een resource. Het routeren naar het juiste stelselonderdeel is afhankelijk van het type resource. In geval van gebruikerstoepassingen wordt routeren door het Omgevingsloket afgehandeld. In geval van een API of service wordt het routeren door het Stelselknooppunt DSO afgehandeld. Technische stelselcomponenten die zowel een gebruikerstoepassing bieden als een API en/of service worden afhankelijk van het resource type door het Omgevingsloket of het Stelselknooppunt DSO afgehandeld.



|  |  |
| --- | --- |
| ☝ | 1. Slechts één stelselcomponent is verantwoordelijke voor het aanbieden van een resource |
| Er is altijd één stelselcomponent verantwoordelijk voor het aanbieden van een resource. Het routeren naar het juiste stelselonderdeel is afhankelijk van het type resource. |

De volgende stelselonderdelen zijn onderkend:

|  |  |
| --- | --- |
| Onderdeel[[7]](#footnote-7) | Toelichting |
| checken | Gebruikerstoepassing Checken. |
| oriënteren | Gebruikerstoepassing Oriënteren. |
| indienen | Gebruikerstoepassing Indienen. |
| catalogus | De Stelselcatalogus DSO. |
| samenwerken-besluiten | Samenwerken aan plannen. |
| samenwerken-aanvragen | Samenwerken aan aanvragen. |

|  |  |
| --- | --- |
| ☝ | 1. Het Omgevingsloket zorgt voor de routering van inhoud (pagina’s) |
| In geval van gebruikerstoepassingen wordt routeren door het Omgevingsloket afgehandeld. |

|  |  |
| --- | --- |
| ☝ | 1. Het Stelselknooppunt zorgt voor de routering van service-verzoeken |
| Het Stelselknooppunt DSO zorgt voor de routering op API- en service-verzoeken naar het juiste stelselonderdeel. |

|  |
| --- |
| 1. De betrokkenheid van het Stelselknooppunt DSO bij het routeren van alle service-verzoeken is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor de eis URI-31. |

### Beveiligingscontext

Het is een best practice dat een component aparte afhandeling biedt voor functionaliteiten die alleen beschikbaar zijn voor specifieke gebruikersrollen die specifieke beveiligingseisen stellen. Bijvoorbeeld de toepassing zelf en beheerfunctionaliteit hiervan.Hiervoor wordt een beveiligings­context toegevoegd aan het pad. Op basis hiervan kan de beveiliging aan de rand van het stelsel afgehandeld worden, zodat alleen geautoriseerde verzoeken het stelsel binnen komen.

|  |  |
| --- | --- |
| ☝ | 1. URI’s definiëren optioneel een expliciete beveiligingscontext |
| In de beveiligingscontext kan bijvoorbeeld onderscheid worden gemaakt tussen “beheer” en “standaardgebruik”, maar ook een veel fijnmazigere indeling is denkbaar. Het is aan de stelselbeheerorganisatie om voor stelselcomponenten de granulariteit in te vullen, afgestemd op de beoogde autorisatie. De afweging hierbij dient te zijn dat een afnemer alleen die functionaliteit of resources zit waar deze toegang toe heeft. |

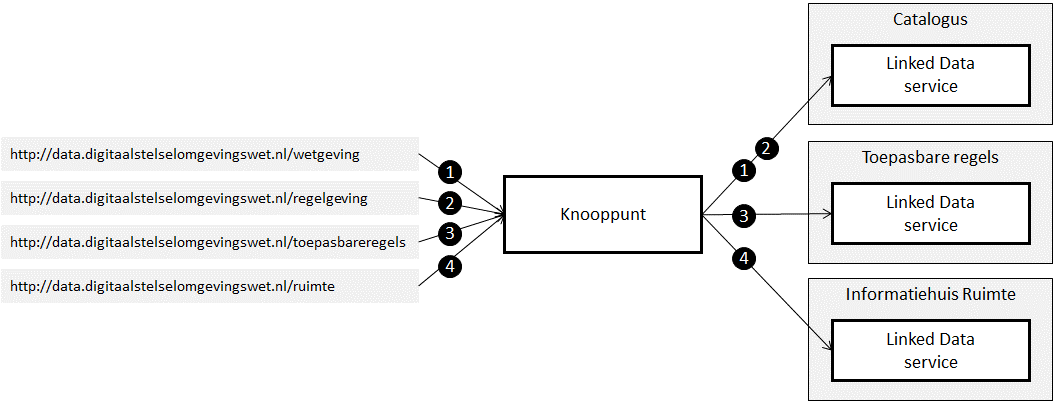
### Domein

Het domein identificeert het DSO-domein waartoe een resource behoort. Een DSO-domein is een functionele identificatie en niet de identificatie van een (technische) stelselcomponent. De volgende DSO-domeinen zijn onderkend:

| Onderdeel | Toelichting |
| --- | --- |
| wetgeving | Resources uit de Omgevingswet en andere wetten. |
| regelgeving | Resources uit Algemene Maatregel van Bestuur (AMvB), Ministeriële regelingen (MR) en lokale regelgeving. |
| toepasbare-regels | Juridische regels omgezet naar begrijpelijke vragen en toegepast in de vorm vragenbomen en formulieren. |
| content | Content resources zoals tekst, helpteksten, plaatjes, video’s, etc. |
| ruimte | Resources van het Informatiehuis[[8]](#footnote-8) Ruimte. |
| bouw | Resources van het Informatiehuis Bouw. |
| lucht | Resources van het Informatiehuis Lucht. |
| afval | Resources van het Informatiehuis Afval. |
| bodem-ondergrond | Resources van het Informatiehuis Bodem en Ondergrond. |
| natuur | Resources van het Informatiehuis Natuur. |
| water | Resources van het Informatiehuis Water. |
| externe-veiligheid | Resources van het Informatiehuis Externe Veiligheid. |
| cultureel-erfgoed | Resources van het Informatiehuis Cultureel Erfgoed. |

|  |  |
| --- | --- |
| ☝ | 1. Een domein binnen een URI is een strikt functionele identificatie |
| Het domein identificeert het DSO-domein waartoe een resource behoort. Een DSO-domein is een functionele identificatie en niet de identificatie van een (technische) stelselcomponent. |

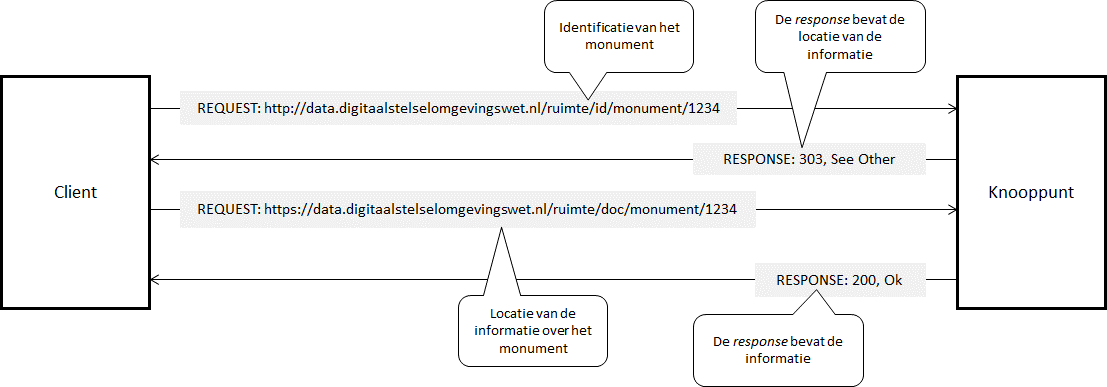
|  |  |
| --- | --- |
| ☝ | 1. Het Stelselknooppunt zorgt voor de routering op basis van domeinen |
| Het Stelselknooppunt DSO zorgt voor de routering van een domein naar de juiste (technische) stelselcomponent. |

In de onderstaande figuur is de domeinroutering door het Stelselknooppunt DSO weergegeven.

|  |
| --- |
| 1. De betrokkenheid van het Stelselknooppunt DSO bij het routeren op basis van domeinen is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor URI-34 en de opbouw van Linked Data URI’s. |

### Type (Linked Data)

Er wordt onderscheid gemaakt tussen de identificatie van de locatie van een resource zelf (URL) om deze op te vragen en de identificatie van de informatie van deze resource (URI). Zo verschilt de URI van een monument van de URL waar de informatie over dit monument is te vinden. De internet­standaarden bieden mechanismen om op basis van de identificerende URI door te verwijzen naar de URL waar deze opgevraagd kan worden. Dit is volledig transparant voor de afnemer.



In bovenstaand voorbeeld is zichtbaar dat de *identificatie* van de resource, conform de standaard, op basis van HTTP plaatsvindt, terwijl de informatie zelf beveiligd op basis van HTTPS wordt geleverd.

Een resource kan op twee manieren opgevraagd worden:

1. http://../ruimte/id/monument/1234
2. https://../ruimte/resource?subject=http://../ruimte/id/monument/1234

Methode (1) betreft de daadwerkelijke identificatie van de resource, die hiermee ook direct de locatie is waarlangs informatie over de resource is op te vragen. Bij methode (2) is sprake van een REST-API, met als parameter de URI van resource. Deze tweede methode maakt het (ook) mogelijk dat de informatie die opgevraagd wordt over een resource geleverd wordt vanuit een ander domein.

Een concept dat gemunt is door de component Toepasbare regels en bekend is in de Catalogus kan op de twee volgende manieren opgevraagd worden:

1. http://../toepasbareregels/id/concept/ExploiterenJachthaven
2. https://.../catalogus/resource?subject=http://../toepasbareregels/id/concept/ExploiterenJachthaven

### Collectie en referentie

De *collectie* geeft de context van de referentie. Vaak wordt hiervoor de naam van een tabel gebruikt. De *referentie* is de identificatie van de betreffende resource in de originele database. Vaak de primaire sleutel in de database.

<domein>[“/”lokaal]”/”<type>“/”<concept>“/”<reference>

Als voorbeeld, de volgende URI identificeert het Kadastrale middelpunt van Nederland, de Onze Lieve Vrouwe Toren in Amersfoort:

http://data.digitaalomgevingsloket.nl/ruimte/id/gebouw/103018712

Bij het opstellen van URI’s voor informatie gelden de volgende afspraken:

* Voor <collectie> wordt een zo generiek mogelijke term gebruikt om onnodig wijzigen van de identificatie te voorkomen. Er is daarom gekozen voor “gebouwen” en niet “torens”.
* Hergebruik bestaande identificaties. Voor de <referentie> wordt de originele identificatie (103018712) van de toren in de database gekozen.

Voor begrippen wordt een vergelijkbare strategie gevolgd, alleen zal hier de <referentie> gelijk zijn aan de term waaronder het begrip bekend is:

http://data.digitaalomgevingsloket.nl/wetgeving/id/concept/Gebouw

## Tijdreizen

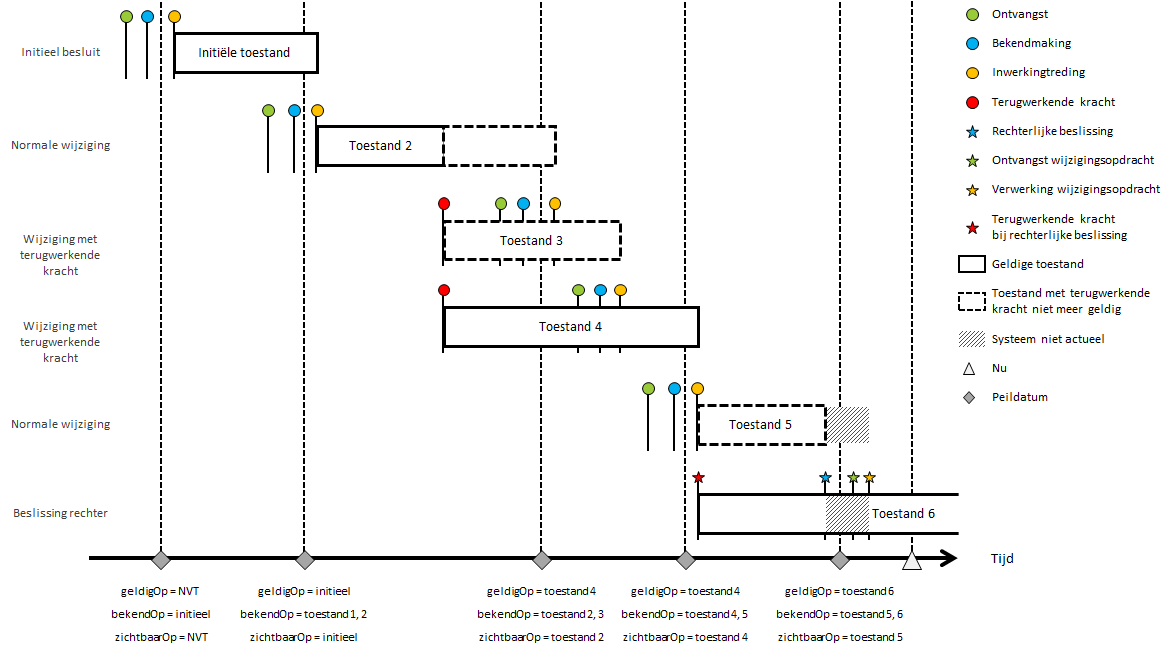
Naast de afspraken over het opstellen van URI’s binnen het DSO, dient rekening te worden gehouden met het feit dat informatie over een resource kan veranderen. Het moet daarom mogelijk zijn om informatie op te vragen over een bepaald moment in de tijd. Voor dit thema, het zogenaamde tijdreizen wordt een afzonderlijke kaderstellende notitie opgesteld.

|  |
| --- |
| 1. De kaderstellende notitie voor het onderwerp tijdreizen is nog in bewerking en wordt afgestemd met PR06, PR13, PR16 en PR30. De gestandaardiseerde query-parameters zullen hoogstwaarschijnlijk niet veranderen. |

Het basisprincipe voor het tijdreizen i.r.t. de URI-strategie is daarbij als volgt:

1. Indien er *geen* specifieke datum wordt meegegeven, dan wordt de *meest actueel geldige* informatie teruggegeven, zoals gebruikelijk op het internet. Indien er nog geen enkele geldige versie bestaat, dan wordt de meest actueel bekende versie teruggegeven;
2. Indien een specifieke datum wordt meegegeven, wordt dit meegegeven als onderdeel van de <query> component.
3. Bij het opgeven van een specifieke datum wordt onderscheid gemaakt tussen:
   1. Wat was de geldende situatie op het aangegeven tijdstip (geldigOp);
   2. Wat was bekend op het aangegeven tijdstip (bekendOp). Dit kan nul, één of meer resultaten opleveren;
   3. Wat was zichtbaar op het aangegeven tijdstip (zichtbaarOp).

|  |  |
| --- | --- |
| ☝ | 1. Tijdreizen wordt ondersteund via gestandaardiseerde query-parameters |
| De volgende standaard query-parameters worden gebruikt om tijdreizen te ondersteunen:   * <geen> meest actueel geldige informatie of de meest actueel   bekende versie * geldigOp=jjjj-mm-dd de geldende situatie op het aangegeven tijdstip * bekendOp=jjjj-mm-dd bekend op het aangegeven tijdstip * zichtbaarOp=jjjj-mm-dd zichtbaar op het aangegeven tijdstip |



Onderstaand voorbeeld geeft de informatie over het begrip “Activiteit” uit de omgevingswet, waarbij de benoemde parameters worden gebruikt:

1. http://data.digitaalstelselomgevingswet/wetgeving/id/concept/Activiteit
2. http://data.digitaalstelselomgevingswet/wetgeving/id/concept/Activiteit?geldigOp=2016-01-01
3. http://data.digitaalstelselomgevingswet/wetgeving/id/concept/Activiteit?bekendOp=2016-01-01
4. http://data.digitaalstelselomgevingswet/wetgeving/id/concept/Activiteit?zichtbaarOp=2016-01-01

Ad 1) Deze URI resulteert in de informatie zoals deze nu actueel geldig is. Nu nog niet in werking is getreden, wordt de meet actueel bekende versie teruggegeven (de geconsolideerde versie is de doorlopende tekst opgebouwd uit het initiële besluit en alle daaropvolgende besluiten).

Ad 2) Deze URI resulteert in de informatie zoals deze geldig is op de genoemde datum. Als het besluit op de aangegeven peildatum nog niet in werking is getreden, leidt dit tot een 404 (niet gevonden): er is dan nog geen geldig begrip “Activiteit”.

Ad 3) Deze URI resulteert in de informatie zoals deze bekend is op de aangegeven peildatum. Als het besluit al wel bekend is gemaakt (in de Staatscourant), resulteert dit in de bekendgemaakte informatie over het betreffende begrip.

Ad 4) Deze URI resulteert in de informatie zoals deze *zichtbaar* was op de aangegeven peildatum. Als er geen wijzigingen op het aangegeven peildatum waren dan is [4] gelijk aan [2] en [3]. Als er wel wijzigingen op het aangegeven peildatum bekend waren dan kan [4] afwijken van [2] en [3].

## Versionering van SOAP web services en REST API’s[[9]](#footnote-9)

Wijzigingen aan het koppelvlak van SOAP web services en REST API’s zijn onvermijdelijk door voortschrijdend inzicht en kennis hoe een systeem opgezet moet worden. De uitdaging is de impact van dit soort wijzigingen te beperken en te zorgen dat bestaande koppelingen blijven werken.

Er zijn een verschillende strategieën die gebruikt kunnen worden om te versioneren. Deze strategieën zijn toepasbare op zowel SOAP web services als REST API’s.

1. Een versienummer toevoegen aan de URI;
2. Een versienummer toevoegen aan de payload;
3. Een eigen request header gebruiken; of
4. Toevoegen aan de HTTP Accept Header.

Het toevoegen aan de URI is de meest rechttoe rechtaan oplossing die breed wordt toegepast en ook voor het stelsel wordt gehanteerd.

## Gewenst formaat

Een URI of URL beschrijft niet het formaat waarin informatie wordt teruggegeven. Conform de internetstandaarden wordt hiervoor de http-accept header gebruikt om het gewenst formaat aan te geven. Dit mechanisme heet “content negotiation”: op basis van het *mime-type* in de http-accept header geeft de afnemer aan in welk formaat de informatie teruggegeven moet worden. Stelselcomponenten ondersteunen content negotiation. Zie voor meer informatie tevens het API Strategie document.

Voor Linked Data resources dient minimaal invulling gegeven te worden aan de volgende mime-types:

| Mime-type | Toelichting |
| --- | --- |
| text/html | Een html weergave. |
| application/ld+json | Een JSON serialisatie van de (linked) data. |
| application/rdf+xml | Een XML serialisatie van de (linked) data. |
| text/turtle | Een turtle serialisatie van de (linked) data. |

Voor niet-linked data varianten (bijvoorbeeld HTML) wordt door middel van link headers (Link) aangegeven welke mime-types geleverd kunnen worden en welke URL’s daarvoor gebruikt moeten worden.

Voor API’s dient minimaal invulling gegeven te worden aan de volgende mime-types:

|  |  |
| --- | --- |
| Mime-type | Toelichting |
| application/json | Een JSON serialisatie van de data (GeoJSON in geval van geo data). |
| application/xml | Een XML serialisatie van de data (GML in geval van geo data). |

Merk op: de Linked Data principes zijn ook goed toe te passen op REST API’s zelf. Zoals het teruggeven van het juiste formaat op basis van de accept-header en HATEOAS-gerelateerde functionaliteit. Dit wordt zeker aanbevolen, maar is geen verplichting t.o.v. REST API’s. Nadere uitwerking is te vinden in het API Strategie document. Uitzondering op bovenstaande mime-types betreffen API’s die conform een specifieke standaard zijn opgesteld. Voorbeelden hiervan zijn API’s die voldoen aan de WMS, WFS en WTS standaarden.

|  |
| --- |
| 1. De betrokkenheid van het Stelselknooppunt DSO bij het routeren van OGC-services is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor de opbouw van URI’s voor WMS, WFS en WTS standaarden. |

## Afspraken URI’s

Hieronder worden de afspraken opgesomd waaraan URI’s en stelselcomponenten die deze implementeren moeten voldoen.

|  |  |
| --- | --- |
| ☝ | 1. URI’s voldoen aan de algemene afspraken |
| Bij het opstellen van URI’s gelden de volgende algemene afspraken:   * URI’s zijn opgebouwd volgens een vaste structuur. * URI’s zijn langdurig stabiel, bevatten geen organisatienamen, systeemnamen, projectnamen, servernamen etc. * URI’s zijn voor zowel mensen als machines leesbaar en begrijpbaar. * URI’s zijn opzoekbaar zodat mensen deze kunnen opvragen. * Als URI's worden opgezocht levert dat nuttige informatie op. * URI’s maken gebruik van standaarden (RDF, RDFa, OWL, SPARQL, JSON, JSON-LD, TTL, HTML, XML). * URI’s linken naar andere URI’s, zodat gerelateerde informatie gevonden kan worden. * URI’s ondersteunen tijdreizen. * Informatie is in verschillende formaten beschikbaar op basis content negotiation. * Parameternamen zijn geüniformeerd. * Een URI bestaat geheel uit kleine letters, waarbij verschillende woorden worden gescheiden d.m.v. een koppelteken (“-“). Uitzondering op deze regel is een begrip die uit Upper Camel Case bestaat (zie URI’s voor begrippen hieronder) en de termen uit een vocabulaire. |

|  |  |
| --- | --- |
| ☝ | 1. URI’s voor informatie voldoen aan de daarvoor geldende afspraken |
| Bij het opstellen van URI’s voor informatie gelden de volgende aanvullende afspraken:   * Voor concept wordt een zo generiek mogelijke term gebruikt om onnodig wijzigingen van de identificatie te voorkomen. * Hergebruik bestaande identificaties. Voor de <reference> wordt waar mogelijk de originele identificatie in de database gebruikt. |

|  |  |
| --- | --- |
| ☝ | 1. URI’s voor begrippen voldoen aan de daarvoor geldende afspraken |
| Bij het opstellen van URI’s voor begrippen gelden de volgende aanvullende afspraken:   * Voor begrippen wordt als <reference> de term waaronder het begrip bekend is gebruikt. * Een begrip begint met een hoofdletter, eventuele spaties worden vervangen door opnieuw te beginnen met een hoofdletter (zogenaamde Upper Camel Case) en diakrieten te verwijderen. Bijvoorbeeld: “exploiteren jachthaven” wordt: “ExploiterenJachthaven”. * Indien er meerdere begrippen zijn met dezelfde term, dan wordt de term aangevuld met een underscore (“\_”), gevolgd door de context waarbinnen het begrip geldig is. Bijvoorbeeld “Steiger\_Water” en “Steiger\_Bouw”. De underscore wordt alleen voor dit doeleinde gebruikt, om duidelijk te maken dat het hier gaat om een context. |

# Bijlage A: Afkortingen

| **Afkorting** | **Toelichting** |
| --- | --- |
| API | Application Programming Interface |
| DSO | Digital Stelsel Omgevingswet |
| HTTP | Hypertext Transfer Protocol |
| JSON | Javascript Object Notation |
| RDF | Resource Description Framework |
| REST | Representational State Transfer |
| RFC | Request For Change |
| SOAP | Simple Object Access Protocol |
| TLS | Transport Layer Security |
| URI | Uniform Resource Identifier |
| URL | Uniform Resource Locator |
| URN | Uniform Resource Name |
| XML | Extensible Markup Language |

# Bijlage B: Begrippen

| **Begrip** | **Toelichting** |
| --- | --- |
| Eindpunt | Een verwijzing naar een URI die web verzoeken accepteert. |
| Linked-data | Linked data is een digitale methode voor het publiceren van gestructureerde gegevens. De methode is gebaseerd op de techniek van HTTP-URI's en RDF. |
| Query | Een vraag die een gebruiker invoert in een zoekmachine om zijn of haar informatie­behoeften te bevredigen. |
| RDF | RDF is een standaard van het World Wide Web Consortium (W3C), oorspronkelijk ontworpen als een metadata-model, maar gaandeweg gebruikt als een formaat om gegevens in het algemeen voor te stellen en uit te wisselen. |
| Resource | Dit is een primitieve binnen de web-architectuur en wordt gebruikt in de definitie van de fundamentele elementen. |
| Serialisatie | Het "serieel" opslaan van een gegevensobject in de vorm van een reeks bytes. Binair of als leesbare tekst zoals bijvoorbeeld het geval is bij JSON. |

# Bijlage C: Eisen

| **Eisen** |
| --- |
| [URI-01 De basisopbouw van URI’s volgt internetstandaard RFC3986 8](#_Toc482685791)  [URI-02 In de basisopbouw van URI zijn de onderkende resourcecategorieën duidelijk herkenbaar 9](#_Toc482685792)  [URI-03 De URI van een webpagina identificeert het verantwoordelijke stelselonderdeel 9](#_Toc482685793)  [URI-04 De URI van een webpagina specificeert optioneel een expliciete beveiligingscontext 9](#_Toc482685794)  [URI-05 De URI van een webpagina identificeert de pagina binnen het stelselonderdeel 9](#_Toc482685795)  [URI-06 De URI van een webpagina specificeert optioneel een reeks query-parameters 9](#_Toc482685796)  [URI-07 De URI van een webpagina specificeert optioneel een fragment binnen de pagina 9](#_Toc482685797)  [URI-08 De URI van een SOAP web-service identificeert het verantwoordelijke stelselonderdeel 10](#_Toc482685798)  [URI-09 De URI van een SOAP web-service specificeert optioneel een expliciete beveiligingscontext 10](#_Toc482685799)  [URI-10 De URI van een SOAP web-service specificeert het major versienummer van de service 10](#_Toc482685800)  [URI-11 De URI van een SOAP web-service identificeert de service binnen het stelselonderdeel 10](#_Toc482685801)  [URI-12 De URI van een REST API identificeert het verantwoordelijke stelselonderdeel 10](#_Toc482685802)  [URI-13 De URI van een REST API specificeert optioneel een expliciete beveiligingscontext 10](#_Toc482685803)  [URI-14 De URI van een REST API specificeert het major versienummer van de API 11](#_Toc482685804)  [URI-15 De URI van een REST API identificeert de collectie binnen het stelselonderdeel 11](#_Toc482685805)  [URI-16 De URI van een REST API specificeert optioneel een verwijzing 11](#_Toc482685806)  [URI-17 De URI van een REST API specificeert optioneel een reeks query-parameters 11](#_Toc482685807)  [URI-18 De URI van Linked Data identificeert het DSO-domein van een resource 11](#_Toc482685808)  [URI-19 De URI van Linked Data identificeert optioneel een pad om lokale informatie te onderscheiden 12](#_Toc482685809)  [URI-20 De URI van Linked Data specificeert het type URI 12](#_Toc482685810)  [URI-21 De URI van Linked Data specificeert een vocabulaire 12](#_Toc482685811)  [URI-22 De URI van Linked Data identificeert de collectie binnen de resource 12](#_Toc482685812)  [URI-23 De URI van Linked Data specificeert optioneel een verwijzing 12](#_Toc482685813)  [URI-24 De URI van Linked Data specificeert optioneel een reeks query-parameters 12](#_Toc482685814)  [URI-25 De URI van Linked Data specificeert per vocabulaire tevens een klassenaam of eigenschapsnaam 12](#_Toc482685815)  [URI-26 Omgevingsloket URI’s definiëren de autoriteit binnen het sub-domein www 13](#_Toc482685816)  [URI-27 Stelselknooppunt URI’s definiëren de autoriteit binnen het sub-domein data 13](#_Toc482685817)  [URI-28 URI’s definiëren optioneel de staging-omgeving binnen de autoriteit 13](#_Toc482685818)  [URI-29 Slechts één stelselcomponent is verantwoordelijke voor het aanbieden van een resource 13](#_Toc482685819)  [URI-30 Het Omgevingsloket zorgt voor de routering van inhoud (pagina’s) 14](#_Toc482685820)  [URI-31 Het Stelselknooppunt zorgt voor de routering van service-verzoeken 14](#_Toc482685821)  [URI-32 URI’s definiëren optioneel een expliciete beveiligingscontext 14](#_Toc482685822)  [URI-33 Een domein binnen een URI is een strikt functionele identificatie 15](#_Toc482685823)  [URI-34 Het Stelselknooppunt zorgt voor de routering op basis van domeinen 15](#_Toc482685824)  [URI-35 Tijdreizen wordt ondersteund via gestandaardiseerde query-parameters 17](#_Toc482685825)  [URI-36 URI’s voldoen aan de algemene afspraken 20](#_Toc482685826)  [URI-37 URI’s voor informatie voldoen aan de daarvoor geldende afspraken 20](#_Toc482685827)  [URI-38 URI’s voor begrippen voldoen aan de daarvoor geldende afspraken 20](#_Toc482685828) |

# Bijlage D: Open Einden

|  |
| --- |
| **Backlog** |
| [OE-01 De betrokkenheid van het Stelselknooppunt DSO bij het routeren van alle service-verzoeken is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor de eis URI-31. 14](#_Toc482685829)  [OE-02 De betrokkenheid van het Stelselknooppunt DSO bij het routeren op basis van domeinen is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor URI-34 en de opbouw van Linked Data URI’s. 15](#_Toc482685830)  [OE-03 De kaderstellende notitie voor het onderwerp tijdreizen is nog in bewerking en wordt afgestemd met PR06, PR13, PR16 en PR30. De gestandaardiseerde query-parameters zullen hoogstwaarschijnlijk niet veranderen. 17](#_Toc482685831)  [OE-04 De betrokkenheid van het Stelselknooppunt DSO bij het routeren van OGC-services is onderwerp van discussie. De uitkomst van deze discussie kan gevolgen hebben voor de opbouw van URI’s voor WMS, WFS en WTS standaarden. 20](#_Toc482685832) |

1. Zie: https://nl.wikipedia.org/wiki/Webservice [↑](#footnote-ref-1)
2. Digikoppeling web services (WUS en ebMS) vallen onder deze categorie. [↑](#footnote-ref-2)
3. Zie: https://en.wikipedia.org/wiki/Representational\_state\_transfer [↑](#footnote-ref-3)
4. Zie: http://www.w3.org/DesignIssues/LinkedData.html [↑](#footnote-ref-4)
5. Dit document heeft als scope de URI-strategie. Interactie-gedreven API’s (Hypermedia/HATEOAS) wordt verder uitgewerkt in het document “API-strategie”. [↑](#footnote-ref-5)
6. Zowel de waarde "amersfoort" als de waarde "0307" zijn afkomstig uit Tabel 33 Gemeentetabel, zie: http://publicaties.rvig.nl/Landelijke\_tabellen/Landelijke\_tabellen\_32\_t\_m\_60\_excl\_tabel\_35/Landelijke\_Tabellen\_32\_t\_m\_60\_in\_pdf\_formaat. [↑](#footnote-ref-6)
7. De Landelijke Voorziening Bekendmaken en Beschikbaarstellen (LVBB) is geen stelselcomponent maar een e-overheid bouwsteen. Deze bouwsteen volgt wel de architectuur van het DSO en daarmee ook deze URI-strategie. De LVBB zal wel een eigen <autoriteit> hebben, namelijk “overheid.nl”. [↑](#footnote-ref-7)
8. De informatiehuizen hebben een bijzondere positie in het stelsel. De informatiehuizen zijn wel voor het aanleveren aan het stelsel gehouden aan de architectuur van het stelsel. Dit betekent dat zij ook gehouden zijn aan de URI-strategie in dit document. Dit wordt opgenomen in de Aansluitvoorwaarden Informatie­huizen. [↑](#footnote-ref-8)
9. Zie voor verdere details het document “API-strategie” en de GAS Stelselknooppunt DSO. [↑](#footnote-ref-9)